1 Forelesning 3

1.1 Big O Kjøretid

Variablen n er vanligvis definert som størrelse på input, men den kan defineres til hva som helst. Når vi beregner kjøretid er $f(n)$ max antall opersjoner PCen må gjøre når et program kjøres på en input av størrelse n. Det vil si, velg den verste input av størrelse n. Teoretisk er vi interessert i

$$\sum_{n=0}^{\infty} f(n)$$

Mens praktisk er vi interessert i når $f(n)$ er 10^9 til 10^{12}. Vi har at

$$\begin{gathered} 3+7 \leq 2 \cdot 7 \\ 124+98 \leq 2 \cdot 124 \end{gathered}$$

Dette gir oss følgende formel

$$a+b \leq 2 \cdot max(a,b)$$

Som resulterer i

$$\begin{gathered} O(1)+O(1)=O(1) \\ O(1)+O(n)=O(n) \\ O(n)+O\left(n^2\right)=O\left(n^2\right) \end{gathered}$$

Big O beskriver et sett med funksjoner. n^2+3n og $7n^2$ er i $O\left(n^2\right)$. Alle funksjoner som er mindre er også med, dvs at $3n$ og $7n\log(n)$ er i $O\left(n^2\right)$.

Definisjon 1. En funksjon $f(n)$ er i $O(g(n))$ dersom det finnes konstanter c og N slik at $f(n) < c \cdot g(n)$ for alle $n > N$, eller, skrevet rent matematisk

$$f(n) \in O(g(n)) \Rightarrow f(n) < c \cdot g(n) \, \forall \, n > N$$

1.2 Doubling Rate

Hva skjer med kjøretid når input blir dobbelt så stor?

- $O(1)$ - Ingenting, ikke avhengig av n.
- $O(n)$ - $c \cdot n \rightarrow c \cdot 2n$, dobbelt så lang tid.
- $O\left(n^2\right)$ - $c \cdot n^2 \rightarrow c \cdot (2n)^2$, fire ganger så stort.
- $O\left(n^3\right)$ - $c \cdot n^3 \rightarrow c\,(2n)^3$, 8 ganger så stort.